International Journal of Research and Applications
Jan-Mar © 2014 Transactions; 1(1): 17-22

ISSN (online): 2349-0020
http://ijraonline.com

COMPUTER SCIENCE ORIGINAL ARTICLE

A Ranking Model Framework for Multiple Vertical Search Domains

Dr. Shoban Babu Sriramoju [1] , Ramesh Gadde [2]
[1] Associate Professor, Department of CSE, S.R Engineering College(Autonomous),
[2] Assistant Professor, Department of CSE, Varadha Reddy College of Engineering,
[1, 2] Affiliated to Jawaharlal Nehru Technological University, Hyderabad-506 371
[1] babuack@yahoo.com, [2] gadde.ramesh@gmail.com

ABSTRACT

Huge information can be obtained from vertical search domains. Often the information is very large in such a way that users need to browse further to get the required piece of information. In this context ranking plays an important role. When ranking is required for every domain, it is tedious task to develop ranking algorithms for every domain explicitly. Therefore it is the need of the hour to have a ranking model that can adapt to various domains implicitly. Recently Geng et al. proposed an algorithm that has a ranking model which can adapt to various domains. This avoids the process of writing various algorithms for different domains. In this paper we built a prototype application that implements the algorithm to provide ranking to the search results of various domains. The experimental results reveal that the algorithm is effective and can be used in the real world applications.

Key words – Ranking, domain specific search, and ranking adaptation.

1. INTRODUCTION

Internet has become information super highway which provides plethora of information of various domains. Internet has been around and being used to add more information to it from various quarters of the world. At the same time information retrieval has been very useful from Internet that led people to obtain required information. Vertical domains, of late, are helping people to obtain required information with ease. However, the search engines are returning huge amount of information that make the end users to spend more time to pick the right information that they need. This caused problems to end users. In order to overcome this problem, many ranking models came into existence. The ranking models include SVM [5], [6], RankNet [3], RankBoost [4], LambdaRank [1] and so on. These algorithms helped to obtain information that can help users immediately.

There are search engines that are domain specific which are moved from broad based searches to domain specific searches in order to provide vertical search information to end users. Such search engines are very useful to end users by returning required documents. Thus many search engines came into existence that can serve images, music and video. They act on various documents of different types and formats.

Ranking models are used by broad based search engines with various techniques. They use Term Frequency (TF) for ranking the results. However, the broad based ranking models provide information of any kind with ranking. They are much generalized ones that can't provide domain specific results. When ranking models are required for vertical domains, many models are needed in order to serve all domains. This will be very cumbersome and time consuming to do so. Therefore it is very much required to have an algorithm that can adapt to various domains in the real world.

From the experiment in the real world it is understood that the broad based search engines can provide information that cannot be useful

immediately unless user browses for more accurate information. To overcome this problem, ranking adaptation to new domains is required by the algorithms of classifiers. In this context research was found in the literature as explored in [7], [8], [9], [10], and [11]. However, it is new research for adapting to new domains i.e. ranking model adaptation. Concept drifting [12] and classifier adaptation existed in the literature. The former is related to predicting rankings while the latter deals with binary targets. These classifiers have some problems as they could not adapt to new vertical domains.

Recently Geng et al. [14] proposed a new ranking adaptation algorithm that helped in domain specific search. One algorithm can adapt to various domains and help in all domains instead of labeled data. The effective adaption of ranking models helped the algorithm to serve users who need diverse information from across the domains. Their algorithm addressed all the problems and made the domain specific search successful with a single ranking model as it can adapt to new domains. In this paper we implemented that model and tested it practically. The empirical results are encouraging. The remainder of the paper is structured into sections. Section 2 focuses on review of literature. Section 3 provides information about the new ranking adaptation model. Section 4 gives details about the experiments, results and evaluation while section 5 provides conclusions.

2. RELATED WORK

In the literature many researches were found on the ranking models that helped to rank returned results from search engines. However, many of the researches were focusing on general information retrieval and not specific to domains. When they are specific to domains, they could not be able to adapt to new domains. Language models for retrieval of information [13], [14] and Classical BM25 [15] worked well. When few parameters are to be adjusted and obtain results, these models worked fine. However, they were inadequate to adapt to new domains for the purpose of ranking. For this reason it is understood from the review of literature that there was a need for making a new ranking model which could adapt to different domains spontaneously without causing problems. Before presenting our ranking model adaptation concept, we would like to review the previous models.

Of late, ranking algorithms came into existence that helped information retrieval with ranking that could let users to get more useful results at the top of the search results. The ranking problem was converted to classification problem by some of the algorithms. The examples of such algorithms include LambdaRank [1], ListNet [2], RankNet [3], RankBoost [4], and Ranking SVM [5], [6] etc. These algorithms have single objective that is optimizing search results. Recently Gent et al. built a new ranking model adaptation algorithm that helps in ranking the search results of any domain. This reduced the need for developing a new ranking model for each domain. Some other ranking related models were developed in [8], [7], [16], and [10]. In this paper we implemented the model developed by Geng et al. [14].

RAKING ADAPTATION

In this section we provide details regarding the ranking model that has been implemented in this paper. Before that we describe the problem statement here. Let set of queries to be Q and set of documents to be D. Human annotators label the search results. The other ranking models studied include PageRank [18] and HITS [17]. Estimating ranking is the purpose of these algorithms. The documents retuned by them will be small and they depended on the prior knowledge in the form of labels and training samples.

Ranking Adaptation SVM

In this paper we have an assumption that the ranking target domain and auxiliary domain are smaller and it is quite possible to have a ranking model that can adapt to various domains. The regularization frameworks that are conventional like Neural Networks [20] and SVM [19] have some problem in obtaining results with various domains. This problem was named as ill-posed problem that needs prior assumption and thus not suitable for ranking model adaptation. Therefore we use regularization framework in order to solve this problem elegantly with the help of an adaptive ranking function.

The ranking adaptation model which has been proposed is as follows.

$$\min = \frac{1-\partial}{2}\|f\|^2 + \frac{\partial}{2}\sum_{r=1}^{R} \Theta r \|f - f^a_r\|^2 + C \Sigma \epsilon_{ijk}$$

$$\text{s.t. } f(\varpi(q_i, d_{ij})) - f(\varpi(q_i, d_{ij})) > 1 - \epsilon_{ijk} \quad \epsilon_{ijk} > 0, \text{ for}$$

$$\forall i \in \{1,2,\ldots\ldots,M\}, \forall j \forall k \in \{1,2,\ldots,n(qi)\} \text{ with } yij > yik$$

Adapting to Multiple Domains

The algorithm proposed in this paper can be extended for multiple domains that can be used in the ranking model adaptation. New domains are learned on the fly by the same ranking model in the process of adaptation. Thus the proposed system supports domain specific search with only one ranking model. The multiple domain adaptation can be formulated as follows with an assumption that certain auxiliary functions help the system to adapt to new domains.

$$\min = \frac{1-\partial}{2}\|f\|^2 + \frac{\partial}{2}\sum_{r=1}^{R} \Theta r \|f - f^a_r\|^2 + C \Sigma \epsilon_{ijk}$$

$$\text{s.t. } f(\varpi(q_i, d_{ij})) - f(\varpi(q_i, d_{ij})) > 1 - \epsilon_{ijk} \quad \epsilon_{ijk} > 0, \text{ for}$$

$$\forall i \in \{1,2,\ldots\ldots,M\}, \forall j \forall k \in \{1,2,\ldots,n(qi)\} \text{ with } yij > yik$$

As expected in this paper, the data is from various domains that have different features related to the corresponding domains. The ranking model we implement practically adapts to those features. The usage of the domain specific features and adapt to new domains make this model useful and avoid the necessity of making so many ranking models. We also considered the ranking loss concept into the framework. Document similarities concept is used in the framework. The margin rescaling is also considered as an optimization problem for rescaling the margin violations if any with respect to adaptability. The same is presented as follows.

$$\min = \frac{1-\partial}{2}\|f\|^2 + \frac{\partial}{2}\sum_{r=1}^{R} \Theta r \|f - f^a_r\|^2 + C \Sigma \epsilon_{ijk}$$

$$\text{s.t. } f(\varpi(q_i, d_{ij})) - f(\varpi(q_i, d_{ij})) > 1 - \epsilon_{ijk} - \sigma_{ijk} \quad \epsilon_{ijk} > 0, \text{ for}$$

$$\forall i \in \{1,2,\ldots\ldots,M\}, \forall j \forall k \in \{1,2,\ldots,n(qi)\} \text{ with } yij > yik$$

In the same fashion slack rescaling is formulated as follows:

$$\text{Max } -1/2 \sum_{ijkk}^{n}(\square) \sum_{ijkk}^{n}(\square) \alpha ijk \; \alpha lmn \; X^T_{ijk} X_{lmn}$$

$$+\sum_{ijkk}^{n}(\square)(1-\delta f^a(x_{ijk}))\; \alpha ijk$$

$$\text{s.t. } f(\varpi(q_i, d_{ij})) - f(\varpi(q_i, d_{ij})) > 1 - \epsilon_{ijk} - \sigma_{ijk} \quad \epsilon_{ijk} > 0, \text{ for}$$

$$\forall i \in \{1,2,\ldots\ldots,M\}, \forall j \forall k \in \{1,2,\ldots,n(qi)\} \text{ with } yij > yik$$

3. EXPERIMENTAL RESULTS

We built a prototype application with web based interface to demonstrate the proof of concept. The application is built using Java/JEE platform that used JDBC, Servlets and JSP. The environment includes a PC with 2 GB RAM with Pentium Core 2 Dual processor is used for experiments. We used datasets such as TD2004 and TD 203 which were gathered from Internet sources. The performance of the model is measured using cumulative gain and means average precision. The results are compared with other baseline methods such as Aux-Only, Tar-Only, and Lin-Comb.

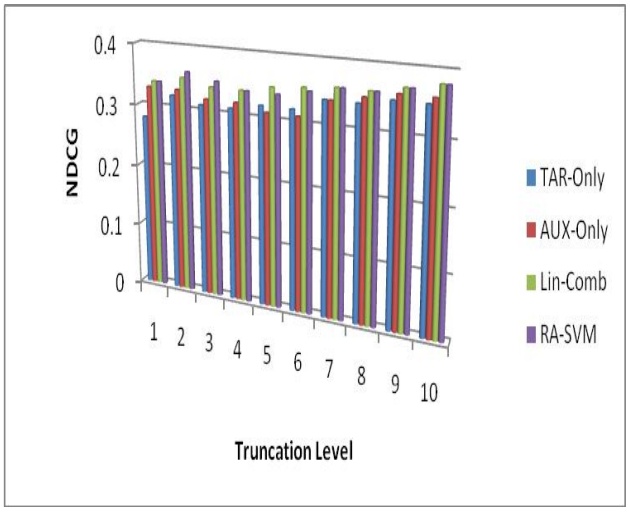

Fig.1.TD200 to TD2004 adaptation with five queries

As can be seen in fig. 1, comparison is made with adaptation performance of proposed algorithm with other three algorithms. The adaptation is from TD2003 dataset to TD2004 dataset with five queries. Out of all, the proposed algorithm has shown best performance. When Aux-Only is compared with Tar-Only, the Aux-Only outperforms the other model.

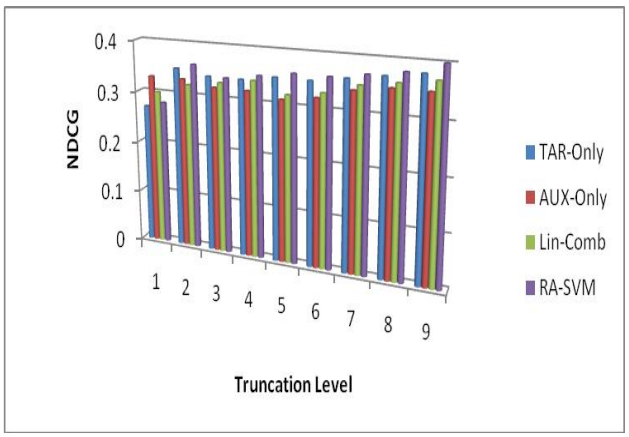

Fig.2.TD2003 to TD2004 adaptation with ten queries

As can be seen in fig. 2, comparison is made with adaptation performance of proposed algorithm with other three algorithms. The adaptation is from TD2003 dataset to TD2004 dataset with ten queries. Out of all, the proposed algorithm has shown best performance. As number of queries is increased, when Aux-Only is compared with Tar-Only, the Tar-Only outperforms the other model.

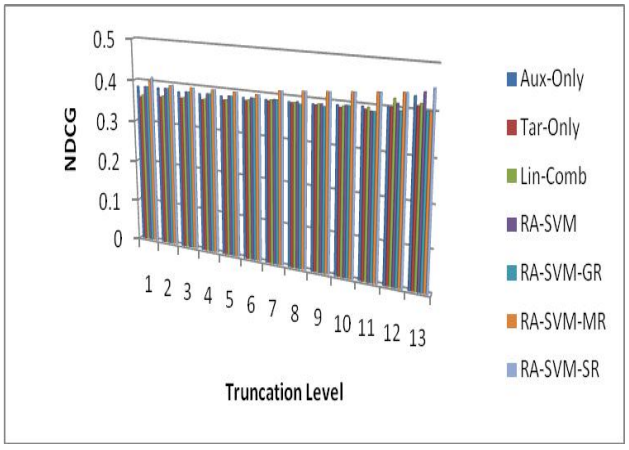

Fig. 3. NDCG Results of web page search to image search adaptation with five labeled queries

As can be seen in fig. 3, adaptation is made from web page search to image search with five labeled queries. The performance of proposed model is compared with other baseline models. The proposed model outperforms other models.

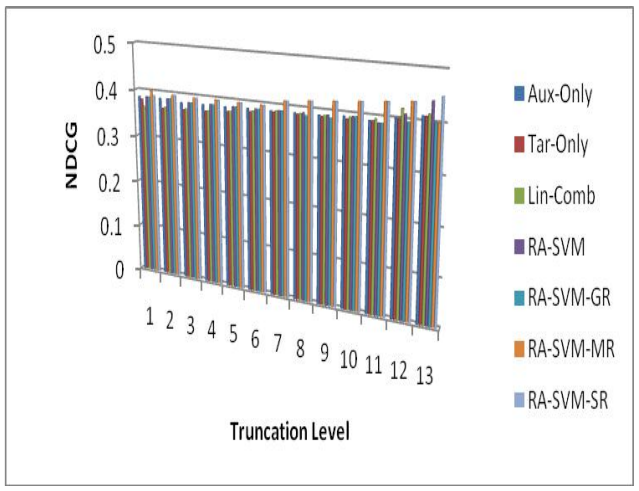

Fig. 4. NDCG Results of web page search to image search adaptation with ten labeled queries

As can be seen in fig. 4, adaptation is made from web page search to image search with ten labeled queries. The performance of proposed model is compared with other baseline models. The proposed model outperforms other models.

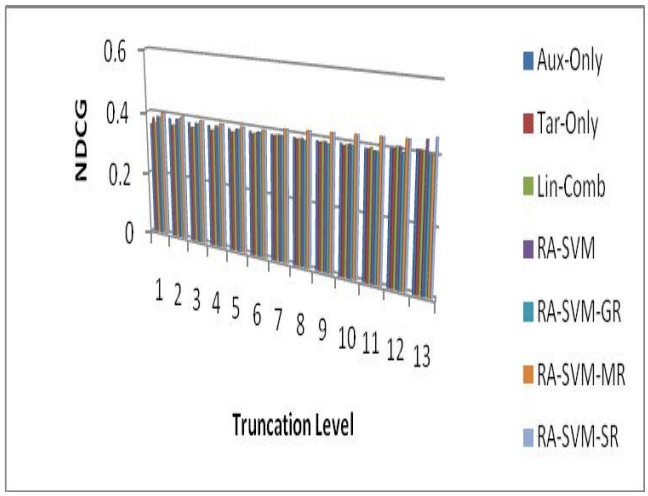

Fig. 5. NDCG Results of web page search to image search adaptation with twenty labeled queries

As can be seen in fig. 5, adaptation is made from web page search to image search with twenty labeled queries. The performance of proposed model is compared with other baseline models. The proposed model outperforms other models.

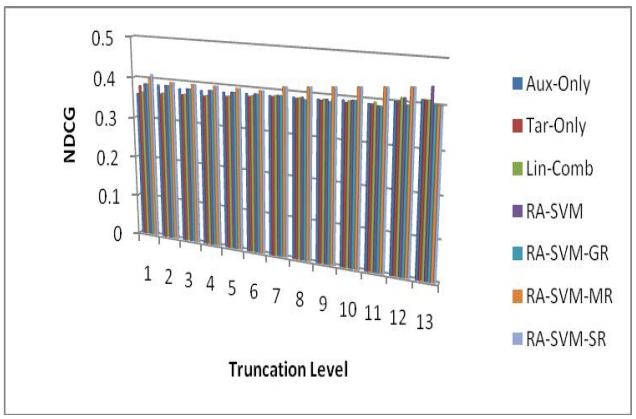

Fig. 6. NDCG Results of web page search to image search adaptation with thirty labeled queries

As can be seen in fig. 6, adaptation is made from web page search to image search with thirty labeled queries. The performance of proposed model is compared with other baseline models. The proposed model outperforms other models.

4. CONCLUSION

In this paper we studied information retrieval systems such as search engines. We came to know that the search engines are broad based search engines and they can't provide domain specific information. However, there are certain domain specific search engines that have problem to adapt to new domains. These search engines provide huge amount of information that makes the user not happy. The reason behind this is that the end user has to browse and spend some time to have exactly required information. This has been improved a lot by various ranking models. However, those ranking models could not adapt to new domains. In this paper we implemented a ranking model that can adapt to new models and thus avoid making many algorithms for each domain. Our model is based on the work done by Geng et al. [14] we built a prototype application that demonstrated the proof of concept. The empirical results are encouraging.

REFERENCES

1. C.J.C. Burges, R. Ragno, and Q.V. Le, "Learning to Rank with Non smooth Cost Functions," Proc. Advances in Neural Information Processing Systems (NIPS '06), pp. 193-200, 2006.

2. Z. Cao and T. Yan Liu, "Learning to Rank: From PairwiseApproach to Listwise Approach," Proc. 24th Int'l Conf. Machine Learning (ICML '07), pp. 129-136, 2007.

3. C.J.C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N.Hamilton, and G. Hullender, "Learning to Rank Using Gradient Descent," Proc. 22th Int'l Conf. Machine Learning (ICML '05), 2005.

4. Y. Freund, R. Iyer, R.E. Schapire, Y. Singer, and G. Dietterich, "An Efficient Boosting Algorithm for Combining Preferences" J. Machine Learning Research, vol. 4, pp. 933-969, 2003.

5. R. Herbrich, T. Graepel, and K. Obermayer, "Large Margin Rank Boundaries for Ordinal Regression,"Advances in Large Margin Classifiers, pp. 115-132, MIT Press, 2000.

6. T. Joachims, "Optimizing Search Engines Using ClickthroughData," Proc. Eighth ACM SIGKDD Int'l Conf. Knowledge Discoveryand Data Mining (KDD '02), pp. 133-142, 2002.

7. J. Blitzer, R. Mcdonald, and F. Pereira, "Domain Adaptation withStructural Correspondence Learning," Proc. Conf. Empirical Methodsin Natural Language Processing (EMNLP '06), pp. 120-128, July2006.

8. W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for Transfer Learning," Proc. 24th Int'l Conf. Machine Learning (ICML '07), pp. 193-200, 2007.

9. H. Shimodaira, "Improving Predictive Inference Under Covariate Shift by Weighting the Log-Likelihood Function," J. Statistical Planning and Inference, vol. 90, no. 18, pp. 227-244, 2000.

10. J. Yang, R. Yan, and A.G. Hauptmann, "Cross-Domain Video Concept Detection Using Adaptive Svms," Proc. 15th Int'l Conf. Multimedia, pp. 188-197, 2007.

11. B. Zadrozny, "Learning and Evaluating Classifiers Under Sample Selection Bias," Proc. 21st Int'l Conf. Machine Learning (ICML '04),p. 114, 2004.

12. R. Klinkenberg and T. Joachims, "Detecting Concept Drift with Support Vector Machines," Proc. 17th Int'l Conf. Machine Learning (ICML '00), pp. 487-494, 2000.

13. J. Lafferty and C. Zhai, "Document Language Models, Query Models, and Risk Minimization for Information Retrieval," Proc. 24th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '01), pp. 111-119, 2001.

14. J.M. Ponte and W.B. Croft, "A Language Modeling Approach to Information Retrieval," Proc. 21st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 275-281, 1998. GENG ET AL.: RANKING MODEL ADAPTATION FOR DOMAIN-SPECIFIC SEARCH 757.

15. S. Robertson and D.A. Hull, "The Trec-9 Filtering Track Final Report," Proc. Ninth Text Retrieval Conf., pp. 25-40, 2000.

16. H. Daume III and D. Marcu, "Domain Adaptation for Statistical Classifiers," J. Artificial Intelligence Research, vol. 26, pp. 101-126, 2006.

17. J.M. Kleinberg, S.R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "The Web as a Graph: Measurements, Models and Methods," Proc. Int'l Conf. Combinatorics and Computing, pp. 1-18, 1999.

18. L. Page, S. Brin, R. Motwani, and T. Winograd, "The Pagerank Citation Ranking: Bringing Order to the Web," technical report, Stanford Univ., 1998.

19. V.N. Vapnik, Statistical Learning Theory. Wiley-Interscience, 1998.

20. F. Girosi, M. Jones, and T. Poggio, "Regularization Theory and Neural Networks Architectures," Neural Computation, vol. 7, pp. 219-269, 1995.